

Amendments to the Claims:

1. (Currently Amended) A computer implemented method for an external program to control or monitor a target software component of an isolated execution unit, the method comprising:

by an external program, creating a new isolated execution unit for execution of the target software component, wherein the external program is outside of the isolated execution unit;

starting an intermediary software component within the isolated execution unit newly-created by the external program, indicating an identifier of the target software component to the intermediary software component;

starting the target software component having the indicated identifier within the isolated execution unit newly-created by the external program;

by the external program indicating one or more parameters representing a desired state for initializing the isolated execution unit;

by the intermediary software component, initializing the isolated execution unit into the desired state based on the indicated one or more parameters; and

establishing a communication path between the intermediary software component and the external program, by the external program, controlling or monitoring target software component, executing in the isolated execution unit newly-created by the external program via the established communication path.

2. (Previously Presented) A computer implemented method as recited in claim 1, wherein the established communication path uses an inter isolation communication protocol.

3. (Previously Presented) A computer implemented method as recited in claim 2, where the inter isolation communication protocol is a remote method invocation technique.

4. (Previously Presented) A computer implemented method as recited in claim 1, wherein the communication path is established by the intermediary software component.

5-8. (Cancelled)

9. (Previously Presented) A computer implemented method as recited in claim 1, further comprising:

indicating an execution control parameter to the intermediary software component;
and

invoking the indicated execution control parameter on the target software component using an application programming interface (API) of the target software component.

10. (Previously Presented) A computer implemented method as recited in claim 9, wherein the execution control parameter is a request that has a first format of an inter isolation communication protocol, the method further comprising translating the first format into a second format that is acceptable by the API of the target software component.

11. (Previously Presented) A computer implemented method as recited in claim 10, wherein the intermediary software component performs the translation.

12. (Previously Presented) A computer implemented method as recited in claim 9, further comprising:

receiving a result at the intermediary software component from the target component in response to the invoked execution control parameter; and

sending the result to the external program.

13. (Previously Presented) A computer implemented method as recited in claim 12, wherein the intermediary software component sends the result.

14. (Previously Presented) A computer implemented method as recited in claim 13, wherein the result has a first format that is acceptable by the API of the target software component, the method further comprising translating the first format into a second format that is an inter isolation communication protocol before sending the result to the external program.

15. (Previously Presented) A computer implemented method as recited in claim 1, wherein the identifier of the target software component is provided by the external program.

16. (Currently Amended) A computer readable medium containing instructions for an external program to control or monitor a target software component of an isolated execution unit, the computer readable medium comprising:

computer code for creating a new isolated execution unit by an external program for execution of the target software component, wherein the external program is outside of the isolated execution unit;

computer code for starting an intermediary software component within the execution unit newly-created by the external program and for indicating an identifier of a target software component to the intermediary software component;

computer code for starting the target software component having the indicated identifier within the isolated execution unit newly-created by the external program;

computer code for indicating by the external program one or more parameters representing a desired state for initializing the isolated execution unit;

computer code for initializing by the intermediary software component the isolated execution unit into the desired state based on the indicated one or more parameters; and

computer code for establishing a communication path between the intermediary software component and the external program, computer code for the external program to control or monitor the target software component, executing in the isolated execution unit newly-created by the external program via the established communication path.

17. (Original) A computer readable medium as recited in claim 16, wherein the established communication path uses an inter isolation communication protocol.

18. (Original) A computer readable medium as recited in claim 17, where the inter isolation communication protocol is a remote method invocation technique.

19. (Original) A computer readable medium as recited in claim 16, wherein the communication path is established by the intermediary software component.

20-23. (Cancelled)

24. (Original) A computer readable medium as recited in claim 16, further comprising:

computer code for indicating an execution control parameter to the intermediary software component; and

computer code for invoking the indicated execution control parameter on the target software component using an application programming interface (API) of the target software component.

25. (Original) A computer readable medium as recited in claim 24, wherein the execution control parameter is a request that has a first format of an inter isolation communication protocol, the computer readable medium further comprising computer code for translating the first format into a second format that is acceptable by the API of the target software component.

26. (Original) A computer readable medium as recited in claim 25, wherein the intermediary software component performs the translation.

27. (Original) A computer readable medium as recited in claim 24, further comprising:
computer code for receiving a result at the intermediary software component from the target component in response to the invoked execution control parameter; and
computer code for sending the result to the external program.

28. (Original) A computer readable medium as recited in claim 27, wherein the intermediary software component sends the result.

29. (Original) A computer readable medium as recited in claim 28, wherein the result has a first format that is acceptable by the API of the target software component, the computer readable medium further comprising computer code for translating the first format into a second format that is an inter isolation communication protocol before sending the result to the external program.

30. (Original) A computer readable medium as recited in claim 16, wherein the identifier of the target software component is provided by the external program.

31. (Currently Amended) A computer implemented system operable to control or monitor a target software component of an isolated execution unit, comprising:
an isolated execution unit created by an external program;
an intermediary software component within the isolated execution unit to initialize the isolated execution unit into a desired state based on one or more parameters indicated by the external program; and
the external program, outside of the isolated execution unit but in the same computer system as the isolated execution unit created by the external program, the external program being configured to indicate an identifier of a target software component to the intermediary software component, wherein the intermediary software component is configured to start the

target software component having the indicated identifier within the isolated execution unit created by the external program and establish a communication path between the intermediary software component and the external program whereby the external program can control or monitor the target software component via the established communication path.

32. (Original) A computer implemented system as recited in claim 31, wherein the established communication path uses an inter isolation communication protocol.

33. (Original) A computer implemented system as recited in claim 32, where the inter isolation communication protocol is a remote method invocation technique.

34-35. (Cancelled)

36. (Original) A computer implemented system as recited in claim 31, wherein the external program is further configured to indicate an execution control parameter to the intermediary software component and the intermediary software component is further configured to invoke the indicated execution control parameter on the target software component using an application programming interface (API) of the target software component.

37. (Original) A computer implemented system as recited in claim 36, wherein the execution control parameter is a request has a first format that is an inter isolation communication protocol, the intermediary software component being further configured to translate the first format into a second format that is acceptable by the API of the target software component.

38. (Original) A computer implemented system as recited in claim 36, wherein the intermediary software component is further configured to receive a result at the intermediary software component from the target component in response to the invoked execution control parameter and send the result to the external program.

39. (Previously Presented) A computer implemented system as recited in claim 38, wherein the result has a first format that is acceptable by the API of the target software component, the intermediary software component being further configured to translate the first format into a second format that is an inter isolation communication protocol before sending the result to the external program.